Data Warehouse And Business Intelligence

CT-472

Course Teacher:

- Dr Umer Farooq

GROUP MEMBERS:

1. Ayan Adnan (DT-22014)
2. Syed Awais Waseen (DT-22033)
3. Muhammad Aman Qazi (DT-22044)
4. Muhammad Aamir (DT-22047)

Data Warehouse And Business Intelligence

CT-472

Data Warehouse Implementation:

FLIGHTS DATAWAREHOUSE AND SYSTEM ANALYTICS

1. Executive Summary

This project implements a comprehensive data warehouse solution for an airline flight management system. The solution integrates data from multiple OLTP databases, external APIs, and CSV files into a centralized Star Schema-based data warehouse (AirlineDW) with interactive Power BI dashboards to enable efficient business intelligence and analytics.

Database Platform: Microsoft SQL Server
ETL Tool: Python (pandas, pyodbc, requests)
Data Sources: 2 OLTP databases, AviationStack API, Customer Satisfaction CSV

2. Project Objectives

Primary Objectives

- **Centralized Data Storage:** Consolidate airline operational data from disparate sources into unified warehouse
- **Business Intelligence Enablement:** Support decision-making through structured reporting and analytics
- **Data Quality Assurance:** Implement validation and cleaning mechanisms for reliable insights
- **Scalable Architecture:** Design for future growth and additional data source integration

Success Metrics

- Successfully integrate 100% of identified data sources
- Achieve data refresh cycles within acceptable timeframes
- Maintain data accuracy and consistency across all dimensions
- Enable real-time business intelligence reporting

Data Warehouse And Business Intelligence

CT-472

3. Data Architecture

3.1 Source Systems

OLTP Database 1: AirFlightsOLTP.OLTP1
Purpose: Customer and booking management

Table	Description	Key Columns
Customers	Customer master data	CustomerID, Name, Contact Info
Bookings	Flight booking records	BookingID, CustomerID, FlightID, BookingDate
Payments	Payment transactions	PaymentID, BookingID, Amount, PaymentMethod

OLTP Database 2: AirFlightsOLTP.OLTP2
Purpose: Flight operations and aircraft management

Table	Description	Key Columns
Aircrafts	Aircraft fleet information	AircraftID, Model, Capacity, ManufactureYear
Flights	Flight schedule and details	FlightID, FlightNumber, DepartureTime, ArrivalTime
Routes	Flight route information	RouteID, Origin, Destination, Distance

External API: AviationStack
Purpose: Real-time flight tracking data
Endpoint: http://api.aviationstack.com/v1/flights
Data Format: JSON
Refresh Rate: On-demand with rate limiting (100 requests/hour)

CSV File: Customer Satisfaction Survey
Purpose: Customer feedback and service ratings
Location: ~/Desktop/AIRFLIGHTSOLTP/test.csv
Contains: 25 columns including satisfaction scores across 14 service dimensions

Data Warehouse And Business Intelligence

CT-472

3.2 Data Warehouse Design

Star Schema Architecture

Fact Table:

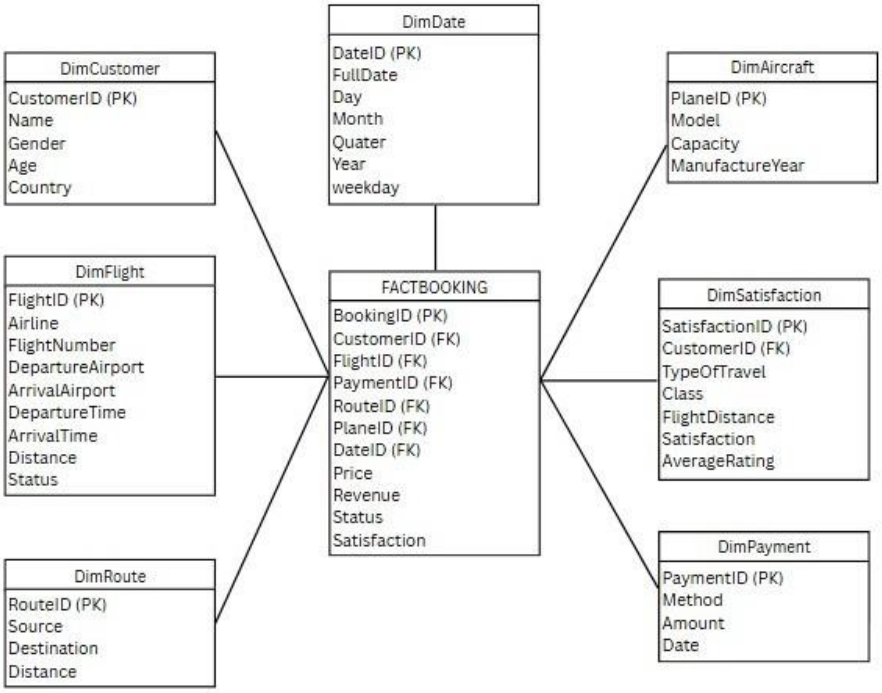
- FactBooking: Central fact table containing booking transactions with foreign keys to all dimensions

Dimension Tables:

- DimCustomer: Customer demographics and information
- DimAircraft: Aircraft specifications and details
- DimRoute: Flight route information
- DimFlight: Flight schedule and operational data
- DimPayment: Payment method and transaction details
- DimSatisfaction: Customer satisfaction metrics and ratings
- DimDate: Time dimension for temporal analysis

Staging Tables:

- Stg_Customers
- Stg_Bookings
- Stg_Payments
- Stg_Aircrafts
- Stg_Flights
- Stg_Routes
- Stg_CustomerSatisfaction
- Stg_API_Flights



4. ETL Pipeline Implementation

4.1 Extract Phase

OLTP Extraction:

- Connect to SQL Server using ODBC Driver 17
- Execute SELECT queries on source tables
- Load data into pandas DataFrames
- Handle connection errors and timeouts

API Extraction:

- HTTP GET request to AviationStack API
- Authentication via access key
- JSON response normalization
- Error handling for rate limits (429) and network issues
- Graceful degradation if API unavailable

CSV Extraction:

- Read CSV file from local filesystem
- Parse 25 columns with mixed data types
- Handle missing values and malformed rows

4.2 Transform Phase

Data Quality Operations:

1. **Data Cleansing:**

- Standardize text fields (e.g., satisfaction values to Title case)
- Remove duplicate records
- Handle null/missing values

2. **Data Enrichment:**

Calculate AverageRating across 14 service dimensions:

- Inflight wifi service
- Departure/Arrival time convenient
- Ease of Online booking
- Gate location
- Food and drink
- Online boarding
- Seat comfort
- Inflight entertainment
- On-board service
- Leg room service
- Baggage handling
- Checkin service
- Inflight service
- Cleanliness

3. Data Validation:

- Verify data type consistency
- Check referential integrity constraints
- Validate business rules

4.3 Load Phase

Two-Stage Loading Process:

Stage 1: Staging Layer

- Truncate existing staging tables
- Bulk insert transformed data
- Maintain full history for audit trail

Stage 2: Dimension & Fact Tables

- Use MERGE statements for Type 1 Slowly Changing Dimensions
- Insert new records
- Update existing records where applicable
- Maintain referential integrity

5. Technical Implementation

5.1 Technology Stack

- Database: Microsoft SQL Server (localhost)
- Programming Language: Python 3.12

Key Libraries:

- pandas: Data manipulation and transformation
- pyodbc: SQL Server connectivity
- requests: API communication
- urllib: URL encoding

5.2 Configuration

SERVER = 'localhost'

DW_DB = 'AirlineDW'

CONN_STR = "DRIVER={ODBC Driver 17 for SQL Server};SERVER=localhost;DATABASE=AirlineDW;Trusted_Connection=yes;"

API_KEY = [Configured]

5.3 Error Handling Strategy

1. API Failures: Return empty DataFrame, skip API staging, continue pipeline
2. Database Connectivity: Log error and terminate with informative message
3. Data Quality Issues: Log warnings, apply default transformations
4. File Access Errors: Validate file path and permissions before processing

5.4 Automated Scheduling

Windows Task Scheduler Implementation:

The ETL pipeline has been automated using Windows Task Scheduler to run at scheduled intervals without manual intervention.

Configuration Details:

- Trigger: Scheduled daily execution (configurable frequency)

- Action: Execute Python script via command line
- Working Directory: Project folder path
- User Account: Configured with appropriate database permissions
- Run Conditions: Run whether user is logged in or not

Task Scheduler Settings:

Task Name: AirlineDW_ETL_Pipeline

Trigger: Daily at [specified time]

Action: python.exe "C:\Users\Hp\Desktop\AIRFLIGHTSOLTP\etl_pipeline.py"

Start in: C:\Users\Hp\Desktop\AIRFLIGHTSOLTP

Run with highest privileges: Yes

Benefits:

- Ensures regular data refresh without manual intervention
- Maintains data currency for Power BI dashboards
- Reduces operational overhead
- Provides consistent data availability for business users

Monitoring:

- Task execution history logged in Task Scheduler
- Python script generates timestamped log files
- Email notifications can be configured for failures

6. Data Quality & Validation

Quality Assurance Measures

1. Source Data Validation:

- Column existence verification
- Data type checking
- Range validation for numeric fields

2. Transformation Validation:

- Pre/post row count reconciliation
- Aggregate value verification
- Duplicate detection

3. Load Validation:

- Foreign key constraint verification
- Primary key uniqueness
- Not-null constraint checks

Data Refresh Strategy

- Staging Tables: Full refresh on each ETL run
- Dimension Tables: MERGE with UPDATE on match, INSERT on no-match
- Fact Tables: Append-only with duplicate prevention

7. Analytics & Business Intelligence Applications

7.1 Potential Use Cases

Operational Analytics:

- Flight on-time performance tracking
- Aircraft utilization rates
- Route profitability analysis
- Booking trend analysis

Customer Analytics:

- Customer satisfaction scoring
- Service quality improvement identification
- Customer segmentation
- Churn prediction

Financial Analytics:

- Revenue per route

- Payment method preferences
- Booking value trends

7.2 Power BI Dashboard Implementation

Interactive Visualizations:

- Executive summary dashboard with key KPIs
- Flight operations monitoring
- Customer satisfaction analysis
- Revenue and booking trends
- Route performance comparison

Key Metrics Displayed:

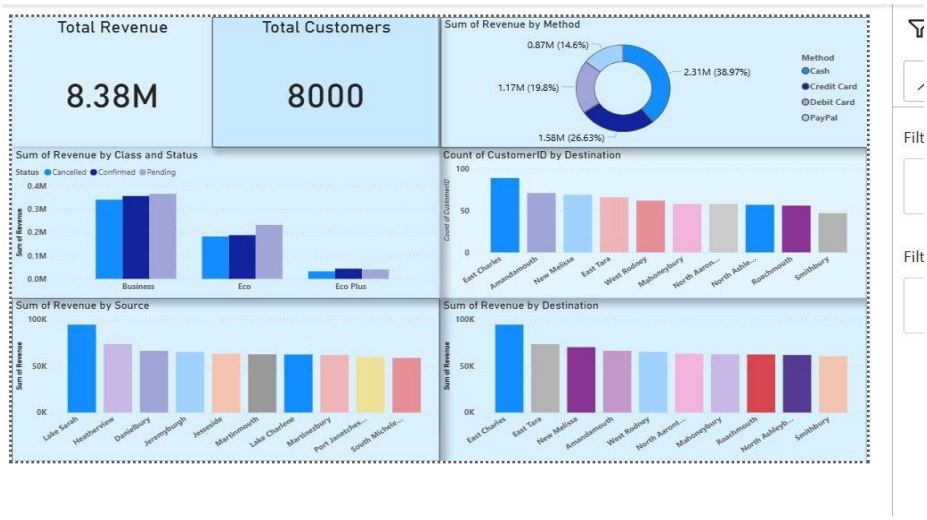
- Total bookings and revenue
- Average customer satisfaction scores
- Aircraft utilization rates
- On-time performance percentages
- Top performing routes
- Payment method distribution

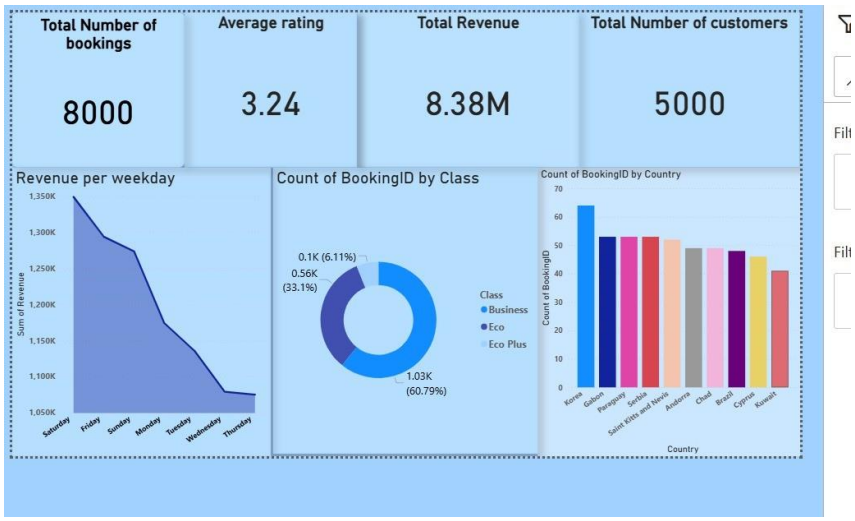
Dashboard Features:

- Real-time data refresh from warehouse
- Interactive filters (date, route, aircraft, customer segment)
- Drill-down capabilities for detailed analysis
- Mobile-responsive design
- Automated report generation

Data Warehouse And Business Intelligence

CT-472



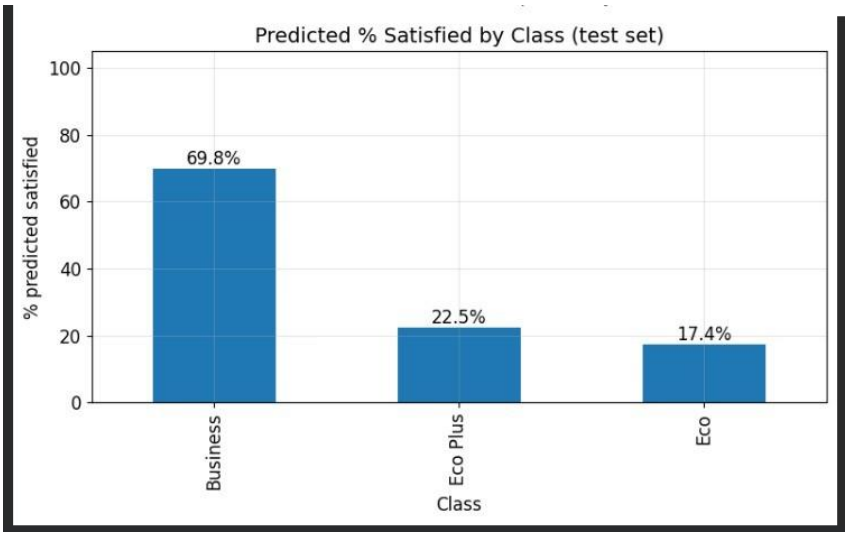


7.3 ML & Data Mining

1. Predicted % Satisfied by Class (Test Set)

Description:

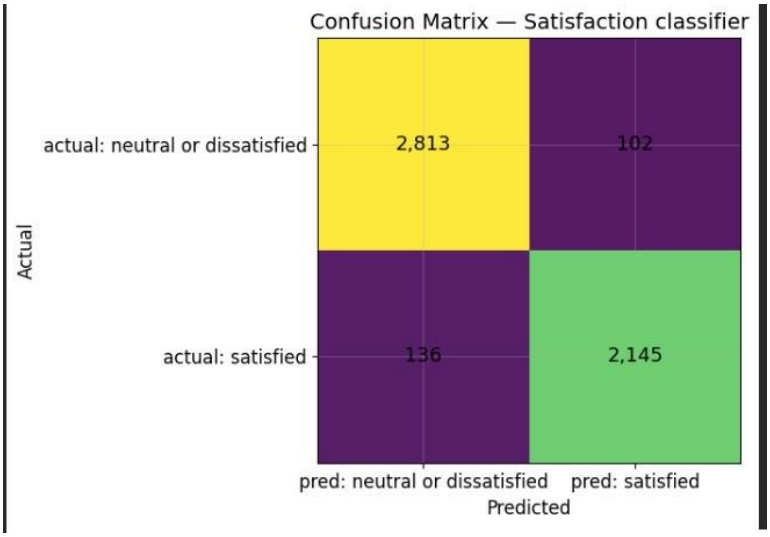
This visualization shows how well the model predicts customer satisfaction for each travel class (Economy, Eco Plus, Business). It highlights which class is expected to have the happiest customers based on model predictions and helps identify service segments that need the most improvement.



2. Confusion Matrix

Description:

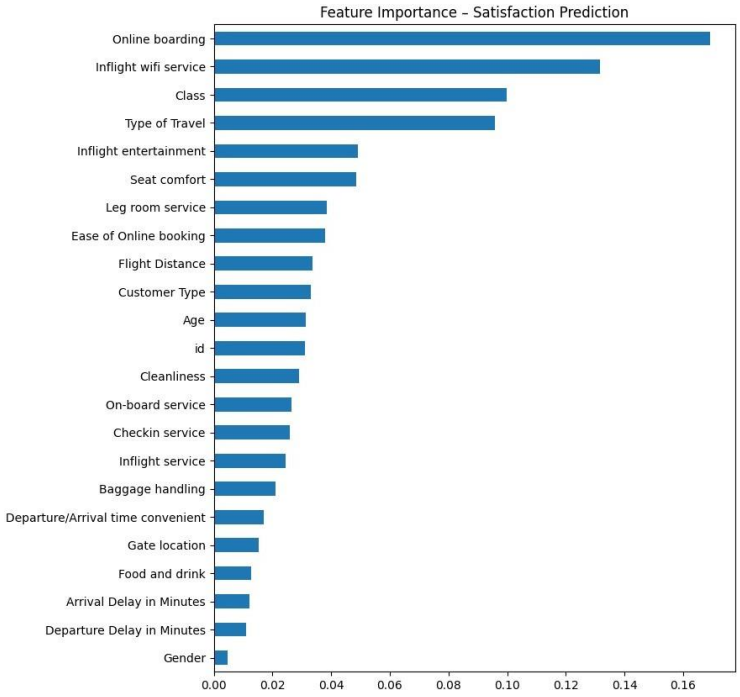
The confusion matrix compares the model's predictions with the actual satisfaction labels. It shows where the model is accurate and where it makes mistakes (false positives and false negatives). This helps evaluate the overall reliability of the satisfaction prediction model.



3. Feature Importance Chart

Description:

This chart ranks the input features by how much they influence the satisfaction prediction. It identifies the key drivers of customer satisfaction—such as inflight service, cleanliness, seat comfort—providing actionable insights for improving the airline's service quality.



8. Benefits Realized

Business Benefits

- Unified Data View: Single source of truth for all airline operations
- Faster Decision Making: Real-time access to integrated data
- Improved Data Quality: Consistent, validated, and reliable data
- Cost Efficiency: Reduced redundant data storage and processing

Technical Benefits

- Scalability: Easy integration of new data sources
- Maintainability: Modular ETL design for easy updates
- Performance: Star schema optimized for analytical queries

- Reliability: Comprehensive error handling and recovery
- Automation: Scheduled execution via Windows Task Scheduler eliminates manual runs

9. Future Enhancements

Planned Improvements

1. Incremental Loading: Implement change data capture for efficiency
2. Data Lineage Tracking: Add metadata tables for audit trail
3. Enhanced Monitoring: Implement comprehensive logging framework with alerts
4. Real-time Streaming: Integrate Kafka for live data feeds
5. Cloud Migration: Move to Azure Synapse or AWS Redshift
6. Enhanced Dashboards: Add predictive analytics visualizations
7. Data Catalog: Implement metadata management system

Monitoring & Maintenance

- Implement logging framework for ETL process monitoring
- Set up alerts for pipeline failures
- Create dashboard for data quality metrics
- Schedule regular performance tuning

10. Conclusion

This data warehouse implementation successfully consolidates airline operational data from multiple heterogeneous sources into a unified, analytics-ready platform. The Star Schema design enables efficient querying and reporting, while the robust ETL pipeline ensures data quality and consistency. The solution provides a solid foundation for business intelligence initiatives and advanced analytics applications.

The modular architecture allows for future scalability and enhancement, positioning the organization for data-driven decision making and competitive advantage in the airline industry.